

### REMARKS

The Applicant respectfully requests further examination and reconsideration in view of the arguments set forth fully below. Claims 1-50 were previously pending in this application. By the above amendment, Claims 1, 13, 25, 37, 47 and 49 have been amended. Within the Office Action, claims 1-50 have been rejected.

#### **Rejections under 35 U.S.C. §102(e)**

Within the Office Action, claims 1, 4-7, 13, 16-19, 25, 28-31, 37, 40-43, and 47 have been rejected under 35 U.S.C. §102(a) as being anticipated by U.S. Patent No. 6,185,569 issued to East et al. (hereafter "East").

The directory tree structure of the present invention is an overlay to a searchable data base. Pointers link positions within the directory tree structure to individual data items. Each node within the directory tree structure includes a plurality of pointers. Since each node is related to a specific category, all data items linked to the node are also related to the specific category. Each node includes a set of parameter fields, and each individual data item linked to the node has a set value for one or more of the parameter fields. In this manner, each individual data item is characterized by a navigation path through the directory tree structure to a node linked to the individual data item, and by its set parameter field values. Accordingly, the pointer that links the node to the individual data item is also characterized by the navigation path to the node and the set parameter field values of the individual data item. East does not teach a pointer that references substantive data according to set parameters.

East teaches a system for verifying the integrity of administrative data for a variety of linked data structure types. One such linked data structure type is described as an index tree, shown in Figure 2 of East as B-tree 200. The B-tree 200 includes a root node 210 and child nodes 211-216. Multiple levels of child nodes can be used. Nodes that have no children are called leaf nodes, nodes 214-216. Only leaf nodes contain substantive data, data fields 290-298. Each node in the data structure includes administrative data in an index field, index fields 270-277 and index fields 280-288. The purpose of the index field is to facilitate the search for a specific item of substantive data in a leaf node. A variety of pointers exist among the nodes including left pointers 220-223, right pointers 230-233, inner child pointers 240-243, forward pointers 250-255, and backward pointers 260-265. A pointer provides a link from one node to another node. A pointer does not link a node to substantive data.

East teaches a distinction between substantive data and administrative data. Substantive data is data stored in a database, whereas administrative data is data relevant to the linked data structure. East teaches that substantive data is stored in data fields, such as data fields 290-298, and that administrative data is stored in index fields, such as index fields 208-288. The pointers of East are references to the linked data structure and therefore point to the index fields containing the administrative data. By way of comparison, the present invention teaches two different types of pointers. A first pointer references nodes within the directory tree structure, the first pointer is specifically referred to as a branch. A second pointer references a specific individual data item, most analogous to the substantive data in East.

Within the Office Action, it is stated that the Applicant argues that East fails to disclose that each node includes a number of parameter fields, where each individual data item associated with the node has a set value for each parameter field. The Applicant contends that using pointers that reference individual data items according to their parameters fields and parameter field values is just one aspect claimed within the independent claims of the present application related to categorizing individual data items according to one or more navigation paths through a directory tree structure and by one or more "set" parameters.

Setting a parameter refers to defining a specific value for a parameter. Each parameter defines a generic field (parameter field) to which a specific value corresponding to the discrete data item is associated. For example, at a "real estate" node, a parameter field name can be "number of rooms" or "price". The parameter field name is different than the actual value eventually associated with the parameter field name in relation to a specific data item. Continuing the example, homes for sale may be described in property fliers. A generic property flier can include many parameters used to describe the home for sale, where each parameter is identified by its parameter field name. The generic property flier can include parameters with parameter field names such as "number of bedrooms", "number of bathrooms", "square footage", "address", and "price". A particular data item associated with the real estate node can be a property flier for a specific three bedroom home for sale. The parameter with parameter field name "number of bedrooms" has a value of "3", in this case, and so on for each of the parameters associated with the property flier. In this manner, it is clear that the value of each parameter, which is specific to a particular data item, is different than the parameter field name of each parameter, which generically defines the type of the parameter. Defining a parameter and a corresponding value of the parameter is commonly referred to as setting a parameter, and the association of the parameter and the particular value is referred to as an attribute-value pair.

By the above amendments, the independent claims have been amended to clarify that each data item within the directory structure is categorized by a navigation path through the directory tree structure and by one or more parameters, where each parameter is set with a corresponding value associated with the data item, thereby forming a set parameter.

Within the Office Action, it is also stated that each node in East includes data in an index field, which is to facilitate the search for a specific item in a leaf node. It is therefore concluded within the Office Action that each pointer references a location of a specific data item in the directory structure. The Applicant respectfully disagrees with this conclusion. As described above, each pointer in East references another pointer. For example, pointer 222 in Figure 2 references node 212 to node 214. However, node 214 includes multiple index fields 280, 281 and 282. The pointer 222 only references the node 214, the pointer 222 does not reference a specific one of the index fields 280, 281, or 282. Therefore, the pointers in East reference nodes. The pointers in East do not reference individual data items.

By way of analogy, if pointers to discrete substantive data items are analogous to mailing addresses of houses, and pointers to nodes are analogous to zip codes, then a pointer to a zip code does not sufficiently identify a specific mailing address for a particular house. The zip code includes many individual mailing addresses, each mailing address analogous to a discrete data item. A pointer to a node (zip code) is not sufficient to identify a discrete data item (mailing address of a house). The present invention specifically teaches a pointer that links a node to an item of data. The present invention also teaches a pointer from one node to another node, such a pointer is referred to as a branch. Therefore, the present invention teaches two distinct types of pointers. One pointer (branches) relates one node to another node, and a second pointer relates a specific node to a discrete data item (pointers). East merely teaches a pointer that relates one node to another node. Although some particular nodes (leaf node 214) in East include a plurality of data fields (data fields 290-292), each data field includes any number of discrete data items. Further, the pointer that specifies a particular node does not also specify a particular discrete data item within the particular node.

As discussed above, East teaches a linked data structure where pointers are used to reference different nodes within the linked data structure. East does not teach a pointer that is categorized by one or more parameters, where each parameter is set with a corresponding value associated with the discrete data item. East also does not teach a pointer that references a discrete data item. Each of these items is explicitly claimed within the independent claims 1, 13, 25, 37, 47, and 49. East does not teach such limitations. East is not designed to define discrete

data items according to their attribute-value pairs (parameter field names and their values), and to then generate a reference structure including pointers that reference each discrete data item according to specified values of predetermined parameters, as in set parameters.

Claim 1 is directed to a method of organizing data within a searchable database. The method of claim 1 comprises the steps of generating a directory tree structure, wherein the directory tree structure includes nodes comprising a designated category for each node and branches comprising links between the nodes, and generating one or more pointers, wherein each pointer corresponds to a specific node and the pointer links the specific node to an item of data within the searchable database, wherein each pointer is categorized by a navigation path through the directory tree structure and by one or more parameters, wherein each parameter is set with a corresponding value associated with an individual data item thereby forming a set parameter, and the parameters are specific to the node in which the pointer is included, further wherein all items of data linked to the specific node by the corresponding pointers of the specific node are related to the designated category of the specific node. As discussed above, East teaches a data structure where pointers point from one node to another node. East does not teach a directory tree structure in which each node includes a collection of pointers, where each pointer links the specific node to an item of data within the searchable database. Further, East does not teach that each node includes a number of parameter fields, where each individual data item associated with the node has a set value for one or more parameter fields. For at least these reasons the independent claim 1 is allowable over the teachings of East.

Claims 4-7 depend on the independent claim 1. As described above, the independent claim 1 is allowable over the teachings of East. Accordingly, claims 4-7 are all also allowable as being dependent on an allowable base claim.

Claim 13 is directed to an organization system for organizing data within a searchable database. The organization system of claim 13 comprises means for generating a directory tree structure, wherein the directory tree structure includes nodes comprising a designated category for each node and branches comprising links between the nodes, and means for generating one or more pointers coupled to the means for generating a directory tree structure, wherein each pointer corresponds to a specific node and the pointer links the specific node to an item of data within the searchable database, wherein each pointer is categorized by a navigation path through the directory tree structure and by one or more parameters, wherein each parameter is set with a corresponding value associated with an individual data item thereby forming a set parameter, and the parameters are specific to the node in which the pointer is included, further wherein all items

of data linked to the specific node by the corresponding pointers of the specific node are related to the designated category of the specific node. As discussed above, East teaches a data structure where pointers point from one node to another node. East does not teach a directory tree structure in which each node includes a collection of pointers, where each pointer links the specific node to an item of data within the searchable database. Further, East does not teach that each node includes a number of parameter fields, where each individual data item associated with the node has a set value for one or more parameter fields. For at least these reasons the independent claim 13 is allowable over the teachings of East.

Claims 16-19 depend on the independent claim 13. As described above, the independent claim 13 is allowable over the teachings of East. Accordingly, claims 16-19 are all also allowable as being dependent on an allowable base claim.

Claim 25 is directed to an organization system for organizing data within a searchable database. The organization system of claim 25 comprises an organization server configured to generate a directory tree structure, wherein the directory tree structure includes nodes comprising a designated category for each node and branches comprising links between the nodes, and to generate one or more pointers, wherein each pointer corresponds to a specific node and the pointer links the specific node to an item of data within the searchable database, wherein each pointer is categorized by a navigation path through the directory tree structure and by one or more parameters, wherein each parameter is set with a corresponding value associated with an individual data item thereby forming a set parameter, and the parameters are specific to the node in which the pointer is included, further wherein all items of data linked to the specific node by the corresponding pointers of the specific node are related to the designated category of the specific node. As discussed above, East does not teach a directory tree structure in which each node includes a collection of pointers, where each pointer links the specific node to an item of data within the searchable database. Further, East does not teach that each node includes a number of parameter fields, where each individual data item associated with the node has a set value for one or more parameter fields. For at least these reasons the independent claim 25 is allowable over the teachings of East.

Claims 28-31 depend on the independent claim 25. As described above, the independent claim 25 is allowable over the teachings of East. Accordingly, claims 28-31 are all also allowable as being dependent on an allowable base claim.

Claim 37 is directed to a network of devices for organizing data within a searchable database. The network of devices of claim 37 comprises one or more computer systems

configured to communicate with other systems, and an organization server configured to couple to the one or more computer systems to generate a directory tree structure, wherein the directory tree structure includes nodes comprising a designated category for each node and branches comprising links between the nodes, and to generate one or more pointers, wherein each pointer corresponds to a specific node and the pointer links the specific node to an item of data within the searchable database, wherein each pointer is categorized by a navigation path through the directory tree structure and by one or more parameters, wherein each parameter is set with a corresponding value associated with an individual data item thereby forming a set parameter, and the parameters are specific to the node in which the pointer is included, further wherein all items of data linked to the specific node by the corresponding pointers of the specific node are related to the designated category of the specific node. As discussed above, East teaches a data structure where pointers point from one node to another node. East does not teach a directory tree structure in which each node includes a collection of pointers, where each pointer links the specific node to an item of data within the searchable database. Further, East does not teach that each node includes a number of parameter fields, where each individual data item associated with the node has a set value for one or more parameter fields. For at least these reasons the independent claim 37 is allowable over the teachings of East.

Claims 40-43 depend on the independent claim 37. As described above, the independent claim 37 is allowable over the teachings of East. Accordingly, claims 40-43 are all also allowable as being dependent on an allowable base claim.

Claim 47 is directed to a method of organizing data within a searchable database. The method of claim 47 comprises generating a directory tree structure, wherein the directory tree structure includes nodes comprising a designated category and an html address for each node and branches comprising links between the nodes, and generating one or more pointers, wherein each pointer corresponds to a specific node and the pointer links the specific node to an item of web-based multimedia within the searchable database, wherein each pointer is categorized by a navigation path through the directory tree structure and by one or more parameters, wherein each parameter is set with a corresponding value associated with an individual web-based multimedia item thereby forming a set parameter, and the parameters are specific to the node in which the pointer is included, further wherein all items of web-based multimedia linked to the specific node by the corresponding pointers of the specific node are related to the designated category of the specific node. As discussed above, East teaches a data structure where pointers point from one node to another node. East does not teach a directory tree structure in which each node includes

a collection of pointers, where each pointer links the specific node to an item of data within the searchable database. Further, East does not teach that each node includes a number of parameter fields, where each individual data item associated with the node has a set value for one or more parameter fields. For at least these reasons the independent claim 47 is allowable over the teachings of East.

**Rejections under 35 U.S.C. §103(a)**

Within the Office Action, claims 2, 3, 8-12, 14, 15, 20-24, 26, 27, 32-36, 38, 39, 44-46, and 48-50 have been rejected under 35 U.S.C. §103(a) as being unpatentable over East in view of U.S. Patent No. 6,292,796 issued to Drucker et al. (hereafter "Drucker").

Drucker teaches an access mechanism that searches current and past literature and selects some or all of the literature for a user, based on criteria established for the user. In a user setup 404, user preferences and profile information are established for a user. The user may also specify search preferences such as the type of literature that is to be searched and the time frame of the search. Additional search criteria is specified using a record link 406, a standing search 408, and an ad hoc access 410. An access mechanism 402 is used to perform filtering after a search is completed. A search is performed on the databases using search criteria obtained from values established in user setup 404, record link 406, standing search 408, and ad hoc access 410. Filtering is then performed on the results of the search query. The search query searches keywords associated with an article, an article's title and/or an article's abstract (Drucker, col. 9, lines 30-32). The user can be notified when a search yields results. In summary, Drucker teaches a keyword search methodology where the search results can be sent to a user using a conventional push technology. However, Drucker does not teach a proactive notification of targeted information, where the information is formatted in a searchable directory tree structure and the information is defined within the directory tree structure according to a navigation path and set parameters, as taught by the present invention.

Drucker teaches searching documents based on a keyword search, and East teaches searching for data using an indexing means. Neither East, Drucker, nor their combination teach determining and tagging documents according to their attribute-value pairs (parameter field names and their values), and to then search for documents within a directory tree structure according to specified values of the predetermined parameters. Further, neither East, Drucker, nor their combination teach a directory tree structure in which each node includes a collection of pointers, where each pointer references a location of a specific data item.

Claim 49 is directed to a method of generating a directory tree structure for organizing data within a searchable database and for accessing the searchable database over the internet. The method of claim 49 comprises the steps of generating one or more nodes wherein each node includes an html address and a designated category, generating links between the nodes wherein each node is linked to at least one other node, further wherein each link is a hypertext link between a first html address of a first node and a second html address of a second node, generating one or more pointers, wherein each pointer corresponds to a specific node and the pointer links the specific node to an item of web-based multimedia within the searchable database, wherein each pointer is categorized by a navigation path through the directory tree structure and by one or more parameters, wherein each parameter is set with a corresponding value associated with an individual web-based multimedia item thereby forming a set parameter, and the parameters are specific to the node in which the pointer is included, further wherein all items of web-based multimedia linked to the specific node by the corresponding pointers of the specific node are related to the designated category of the specific node, and establishing a connection over the internet to the directory tree structure for accessing the searchable database. As discussed above, neither East, Drucker, nor their combination teach a directory tree structure in which each node includes a collection of pointers, where each pointer corresponds to an item of web-based multimedia within the searchable database. Further, neither East, Drucker, nor their combination teach searching for documents within a directory tree structure according to specified values of predetermined parameters. For at least these reasons the independent claim 49 is allowable over the teachings of East, Drucker, and their combination.

Claim 50 depends on the independent claim 49. As described above, the independent claim 49 is allowable over the teachings of East, Drucker, and their combination. Accordingly, claim 50 is also allowable as being dependent on an allowable base claim.

Claims 2, 3 and 8-12 are dependent on the independent claim 1. Claims 14, 15 and 20-24 are dependent on the independent claim 13. Claims 26, 27 and 32-36 are dependent on the independent claim 25. Claims 38, 39 and 44-46 are dependent on the independent claim 37. Claim 48 is dependent on the independent claim 47. As discussed above, the independent claims 1, 13, 25, 37, and 47 are each allowable over the teachings of East. Accordingly, claims 2, 3, 8-12, 14, 15, 20-24, 26, 27, 32-36, 38, 39, 44-46, and 48 are all also each allowable as being dependent on an allowable base claim.



For the reasons given above, Applicant respectfully submits that claims 1-50 are now in a condition for allowance, and allowance at an early date would be appreciated. Should the Examiner have any questions or comments, he/she is encouraged to call the undersigned attorney at (408) 530-9700.

Respectfully submitted,  
HAVERSTOCK & OWENS LLP

Date: December 19, 2003

By: Jonathan O. Owens  
Jonathan O. Owens  
Reg. No. 37,902

Attorneys for Applicant